

One-sided Precoder Designs on Manifolds for Interference Alignment

Chen Zhang, Huarui Yin, and Guo Wei

Department of Electrical Engineering and Information Science

University of Science and Technology of China, HeFei, Anhui, 230027, P.R.China

Email: zhangzc@mail.ustc.edu.cn, {yhr, wei}@ustc.edu.cn

Abstract—Interference alignment (IA) is a technique recently shown to achieve the maximum degrees of freedom (DoF) of K -user interference channel. In this paper, we focus on the precoder designs on manifolds for IA. By restricting the optimization only at the transmitters' side, it will alleviate the overhead induced by alternation between the forward and reverse links significantly. Firstly a classical steepest descent (SD) algorithm in multi-dimensional complex space is proposed to achieve feasible IA. Then we reform the optimization problem on Stiefel manifold, and propose a novel SD algorithm based on this manifold with lower dimensions. Moreover, aiming at further reducing the complexity, Grassmann manifold is introduced to derive corresponding algorithm for reaching the perfect IA. Numerical simulations show that the proposed algorithms on manifolds have better convergence performance and higher system capacity than previous methods, also achieve the maximum DoF.

I. INTRODUCTION

Interference alignment (IA) is a technique recently proposed in [1], [2] and [3], which can achieve much higher wireless networks capacity than previously believed [4]. For the K -user $M \times M$ multiple-input multiple-output (MIMO) interference channel, the sum capacity is

$$C_{sum} = \frac{KM}{2} \log(1 + SNR) + o(\log(SNR)) \quad (1)$$

with achievability of $KM/2$ degrees of freedom (DoF), which is defined as

$$DoF = \lim_{SNR \rightarrow \infty} \frac{C_{sum}}{\log(SNR)} \quad (2)$$

It means each transmitter-receiver pair is able to communicate with achievability DoF of $M/2$, irrespective of the number of interferers.

A feasible way to align interference is to design such a precoder that coordinates transmitting directions, in order to force the interference overlapping at each receiver, and reserve half of the signaling space interference-free for the desired signals. Based on the assumption of channel reciprocity, some previous works such as [5] – [10] iteratively optimize both the precoder matrices and interference suppression filters, by alternating between the forward and reverse links to achieve interference alignment in a distributed way. However, with the assumption of channel reciprocity, the applicability of these algorithms are restricted within TDD systems only. Moreover, alternation between the forward and reverse links needs tight synchronization and feedback at each node, which

may introduce too much overhead when the channel varies quickly. Furthermore, the transmitters and receivers exchange their “roles” during each iteration of optimization. Thus this scheme is inappropriate for the receivers with limited computing ability.

On the other hand, most of the previous works above employ traditional constrained optimization techniques that work in high dimensional complex space. Unavoidably, several shortcomings are accompanied with the traditional constrained optimization techniques such as low-converging speed and high-complexity.

To overcome these limitations, in this paper, we introduce optimization on matrix manifolds into the precoder design for interference alignment, and limit the optimization only at the transmitters' side. Optimization algorithms on manifolds consist the merits of lower complexity and better numerical properties. Firstly for the sake of comparison, by employing classical constrained optimization method, a steepest descent (SD) algorithm in multi-dimensional complex space is provided to design the precoder of interference alignment. Then we reformulate the constrained optimization problem to an unconstrained and non-degraded one on the complex Stiefel manifold with lower dimensions. We locally parameterize the manifold by Euclidean projection from the tangent space onto the manifold instead of the traditional method by moving descent step along the geodesic in [11] and [12]. Thus the SD algorithm on Stiefel manifold is proposed to achieve feasible interference alignment. To further reduce the computation complexity in terms of dimensions of manifold, we explore the unitary invariance property of our cost function, and solve the optimization problem on the complex Grassmann manifold, then present corresponding SD algorithm on Grassmann manifold for interference alignment.

We not only generalize optimization algorithm on manifolds, but also turn the algorithm into an efficient numerical procedure to achieve perfect interference alignment. Moreover by limiting the optimization algorithms performed at the transmitters' side only, all the three proposed algorithms are transparent at the receivers. Additionally, overhead generated by synchronization and feedback no longer exists since only transmitters participate in the iteration. Besides, by relaxing the assumption of channel reciprocity, our algorithms are applicable to both TDD and FDD systems. Furthermore, numerical simulation shows that the novel algorithms on

manifolds have better convergence performance and higher system capacity than previous methods. Finally, we prove the convergence of the proposed algorithms.

The paper is organized as following. Interference channel is briefly summarized in Section II and the mathematical model of interference alignment is presented in Section III, followed by the detailed procedures of all three proposed SD algorithms for interference alignment in Section IV. In section V numerical simulations and corresponding discussion are stated. And the conclusion is given in the last section.

Notation: We use bold uppercase letters for matrices or vectors. \mathbf{X}^T and \mathbf{X}^\dagger denote the transpose and the conjugate transpose (Hermitian) of the matrix \mathbf{X} respectively. Assuming the eigenvalues of a matrix \mathbf{X} and their corresponding eigenvectors are sorted in ascending order, λ_X^i denotes the i^{th} eigenvalue of the matrix \mathbf{X} . Then \mathbf{I} represents the identity matrix. Moreover $\text{tr}(\cdot)$ indicates the trace operation. And the Euclidean norm of \mathbf{X} is $\|\mathbf{X}\| = \sqrt{\text{tr}(\mathbf{X}^\dagger \mathbf{X})}$. $[\mathbf{X}]$ denotes the subspace spanned by the columns of \mathbf{X} . $\mathbb{C}^{n \times p}$ represents the $n \times p$ dimensional complex space assuming $n > p$. \mathbb{R}^+ represents positive real number space. $\Re\{\cdot\}$ and $\Im\{\cdot\}$ denote the real and imaginary parts of a complex quantity, respectively. Finally $\kappa = \{1, \dots, K\}$ is the set of integers from 1 to K .

II. SYSTEM MODEL

Consider the K-user wireless MIMO interference channel depicted in Fig. 1 where each transmitter and receiver are equipped with $M^{[k]}$ and $N^{[k]}$ antennas respectively. Each transmitter communicates with its corresponding receiver, and creates interference to all the other receivers. $d^{[k]}$ is the desired number of data streams between the k^{th} transmitter-receiver pair. Additionally, $\mathbf{H}^{[kj]}$ denotes the $N^{[k]} \times M^{[j]}$ channel coefficients matrix from the j^{th} transmitter to the k^{th} receiver, and is assumed to have i.i.d. complex Gaussian random variables, drawn from a continuous distribution. Finally the received signal vector at receiver k after zero-forcing the interference is denoted by

$$\bar{\mathbf{Y}}^{[k]} = \mathbf{U}^{[k]\dagger} \mathbf{Y}^{[k]} = \mathbf{U}^{[k]\dagger} \left(\sum_{j=1}^K \mathbf{H}^{[kj]} \mathbf{V}^{[j]} \mathbf{S}^{[j]} + \mathbf{W}^{[k]} \right), k \in \kappa \quad (3)$$

where each element of the $d^{[j]} \times 1$ vector $\mathbf{S}^{[j]}$ represents an independently encoded Gaussian symbol with power $P^{[j]}/d^{[j]}$ that beamformed with the corresponding $M^{[j]} \times d^{[j]}$ precoder matrix $\mathbf{V}^{[j]}$, and then transmitted by the transmitter j . $\mathbf{U}^{[k]}$ is the $N^{[k]} \times d^{[k]}$ interference zero-forcing filter at the receiver k . And $\mathbf{W}^{[k]}$ is the i.i.d. complex Gaussian noise with zero mean unit variance.

III. MATHEMATICAL MODEL

A. Feasibility of Interference Alignment

The quality of alignment is measured by the interference power remaining in the intended signal subspace at each

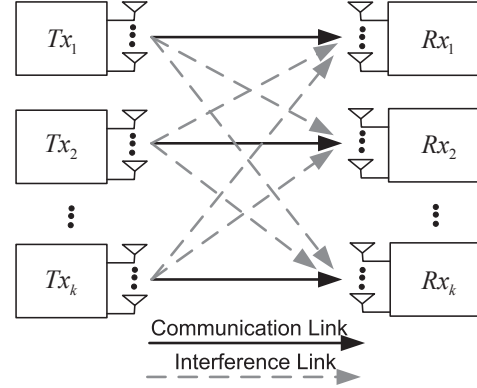


Fig. 1. K-user MIMO interference channel

receiver. Therefore interference alignment can be achieved by progressively reducing the power of leakage interference. And if interference alignment is feasible, the leakage interference can eventually be coordinated to zero. From [5], it can be obtained that the $d^{[k]}$ -dimensional received signal subspace that contains the least interference is the space spanned by the eigenvectors corresponding to the $d^{[k]}$ -smallest eigenvalues of the interference covariance matrix $\mathbf{Q}^{[k]}$. Consequently, we try to minimize the sum of interference power spilled to the desired signal subspaces, by minimizing the sum of the absolute value of $d^{[k]}$ -smallest eigenvalues of the interference covariance matrix at each receiver to create $d^{[k]}$ -dimensional interference-free subspace for the desired signal.

B. Cost Function

As previously stated, we try to minimize the sum of the $d^{[k]}$ -smallest eigenvalues (in absolute value) of the interference covariance matrix at each receiver, over the set of precoder matrices $\mathbf{V}^{[1]}, \dots, \mathbf{V}^{[K]}$ [13]. Therefore, we define the cost function as:

$$\min_{\mathbf{V}^{[1]}, \dots, \mathbf{V}^{[K]}} f = \sum_{k=1}^K \sum_{i=1}^{d^{[k]}} |\lambda_{Q^{[k]}}^i|, k, j \in \kappa \quad (4)$$

subject to $\mathbf{V}^{[j]\dagger} \mathbf{V}^{[j]} = \mathbf{I}_{d^{[j]}}$

where

$$\mathbf{Q}^{[k]} = \sum_{\substack{j=1 \\ j \neq k}}^K \frac{P^{[j]}}{d^{[j]}} \mathbf{H}^{[kj]} \mathbf{V}^{[j]} \mathbf{V}^{[j]\dagger} \mathbf{H}^{[kj]\dagger} \quad (5)$$

is the interference covariance matrix at receiver k . With the assumption that all the eigenvalues are sorted in ascending order, $\lambda_{Q^{[k]}}^i$ represents the i^{th} eigenvalue of the corresponding interference covariance matrix $\mathbf{Q}^{[k]}$. And because $\mathbf{Q}^{[k]}$ is a Hermitian matrix, all its eigenvalues are real. Therefore, the cost function $f(\mathbf{V})$, $f: \mathbb{C}^{n \times p} \rightarrow \mathbb{R}^+$ is built.

IV. ALGORITHMS ON DIFFERENT TOPOLOGIES FOR INTERFERENCE ALIGNMENT

A. The Steepest Descent Algorithm in Complex Space for IA

Since our cost function: $f(\mathbf{V})$, $f : \mathbb{C}^{n \times p} \rightarrow \mathbb{R}^+$ is differentiable, intuitively the steepest descent method can be employed to make the cost function converge to a local optimal point efficiently. As well known, the SD method typically can be iteratively processed as: by adding an update increment to the previous iterate in order to reduce the cost function; the update direction and step size are generally computed using a local model of the cost function, typically based on first derivatives of the cost function. Therefore we will first find the closed-form expression of the steepest descent direction in $\mathbb{C}^{n \times p}$, then employ a property step size rule for each iteration.

As previously stated, the steepest descent method is tightly related to derivative and differentiation. In order to get the derivative of $f(\mathbf{V})$ over \mathbf{V} , two Jacobian matrices blocks are employed as:

$$df = \begin{bmatrix} \mathbf{D}_R^{[1]} & \dots & \mathbf{D}_R^{[K]} \end{bmatrix} \begin{bmatrix} d\mathbf{V}_R^{[1]} \\ \vdots \\ d\mathbf{V}_R^{[K]} \end{bmatrix} + \begin{bmatrix} \mathbf{D}_I^{[1]} & \dots & \mathbf{D}_I^{[K]} \end{bmatrix} \begin{bmatrix} d\mathbf{V}_I^{[1]} \\ \vdots \\ d\mathbf{V}_I^{[K]} \end{bmatrix} \quad (6)$$

where $\mathbf{V}_R^{[j]} = \Re\{\mathbf{V}^{[j]}\}$, and $\mathbf{V}_I^{[j]} = \Im\{\mathbf{V}^{[j]}\}$. $\mathbf{D}_R^{[j]}$ and $\mathbf{D}_I^{[j]}$ are the $d^{[j]} \times M^{[j]}$ Jacobian matrices which denote the partial differential relation of the cost function over the real and imaginary parts of $\mathbf{V}^{[j]}$ respectively. The detail of mathematical derivations can be found in [13] and [14]. Thus, the derivative of f over $\mathbf{V}^{[j]}$ is given by

$$\mathbf{D}_V^{[j]} = (\mathbf{D}_R^{[j]} + i\mathbf{D}_I^{[j]})^T \quad (7)$$

The inner product typically defined in the Euclidean multi-dimensional space is given as:

$$\langle \mathbf{Z}_1, \mathbf{Z}_2 \rangle = \text{tr}(\mathbf{Z}_2^\dagger \mathbf{Z}_1) \quad (8)$$

Then, under the given inner product, the steepest descent direction is:

$$\mathbf{Z}^{[j]} = -\mathbf{D}_V^{[j]} = -(\mathbf{D}_R^{[j]} + i\mathbf{D}_I^{[j]})^T \quad (9)$$

Once the formulation of steepest descent direction $\mathbf{Z}^{[j]}$ is defined in (9), it is necessary to choose a suitable positive step size $\beta^{[j]}$ for each iteration. The Armijo step size rule [15] states that $\beta^{[j]}$ should be chosen to satisfy the following inequalities:

$$f(\mathbf{V}) - f(\mathbf{V} + \beta \mathbf{Z}^{[j]}) \geq \frac{1}{2} \beta^{[j]} \langle \mathbf{Z}^{[j]}, \mathbf{Z}^{[j]} \rangle \quad (10)$$

$$f(\mathbf{V}) - f(\mathbf{V} + 2\beta \mathbf{Z}^{[j]}) < \beta^{[j]} \langle \mathbf{Z}^{[j]}, \mathbf{Z}^{[j]} \rangle \quad (11)$$

Rule (10) guarantees that the step $\beta^{[j]} \mathbf{Z}^{[j]}$ will expressively decrease the cost function, whereas (11) undertakes that the step $2\beta^{[j]} \mathbf{Z}^{[j]}$ would not be a better choice. A direct procedure for acquiring a suitable $\beta^{[j]}$ is to keep on doubling $\beta^{[j]}$ until

(11) no longer holds and then halving $\beta^{[j]}$ until it satisfies (10). It can be proved that such $\beta^{[j]}$ can always be found [16].

Consolidating all the ideas stated above, we present our algorithm in **Algorithm 1**. In Step 3 and Step 4, the Armijo step rule is performed to find a proper convergence step length. Generally speaking, Step 3 ensures the chosen step $\beta^{[j]}$ will significantly reduce the cost function while Step 4 prevents $\beta^{[j]}$ from being too large that may miss the potential optimal point. The operator $gs(\cdot)$ means Gram-Schmidt Orthogonalization [17] of a matrix, which guarantees the newly computed solution $\mathbf{V}^{[j]}$ (or $\mathbf{B}_1^{[j]}, \mathbf{B}_2^{[j]}$) still satisfies the unitary constraint.

Algorithm 1 The Steepest Descent Algorithm in Complex Space for IA

Start with arbitrary precoder matrices $\mathbf{V}^{[1]}, \dots, \mathbf{V}^{[K]}$, set initial step size $\beta^{[j]} = 1$ and begin iteration.

for $j = 1 \dots K$

(1) Compute the Jacobian matrix $\mathbf{D}_R^{[j]}$ and $\mathbf{D}_I^{[j]}$

(2) Get the steepest descent direction

$$\mathbf{Z}^{[j]} = -\mathbf{D}_V^{[j]} = -(\mathbf{D}_R^{[j]} + i\mathbf{D}_I^{[j]})^T$$

(3) Compute $\mathbf{B}_1^{[j]} = gs(\mathbf{V}^{[j]} + 2\beta^{[j]} \mathbf{Z}^{[j]})$,

if $f(\mathbf{V}^{[1]}, \dots, \mathbf{V}^{[K]}) - f(\mathbf{V}^{[1]}, \dots, \mathbf{V}^{[j-1]}, \mathbf{B}_1^{[j]}, \mathbf{V}^{[j+1]}, \dots, \mathbf{V}^{[K]}) \geq \beta^{[j]} \text{tr}(\mathbf{Z}^{[j]\dagger} \mathbf{Z}^{[j]})$, then set $\beta^{[j]} := 2\beta^{[j]}$, repeat Step 3.

(4) Compute $\mathbf{B}_2^{[j]} = gs(\mathbf{V}^{[j]} + \beta^{[j]} \mathbf{Z}^{[j]})$,

if $f(\mathbf{V}^{[1]}, \dots, \mathbf{V}^{[K]}) - f(\mathbf{V}^{[1]}, \dots, \mathbf{V}^{[j-1]}, \mathbf{B}_2^{[j]}, \mathbf{V}^{[j+1]}, \dots, \mathbf{V}^{[K]}) < \frac{1}{2} \beta^{[j]} \text{tr}(\mathbf{Z}^{[j]\dagger} \mathbf{Z}^{[j]})$, then set $\beta^{[j]} := \frac{1}{2} \beta^{[j]}$, repeat Step 4.

(5) $\mathbf{V}^{[j]} = gs(\mathbf{V}^{[j]} + \beta^{[j]} \mathbf{Z}^{[j]})$

(6) Continue till the cost function f is sufficiently small.

Discussion:

- (i) The inner product and the gradient direction are defined in different topologies in [13]. However it is considered to be inappropriate because the gradient is defined after the inner product is given only. In other words, the inner product and the gradient direction must be defined in the same topology. Our proposed algorithm rectifies the topology flaw in [13], thus avoids the risk of non-convergence.
- (ii) It can be concluded that Algorithm 1 belongs to the classical optimization method, which means it works in multi-dimensional space $\mathbb{C}^{n \times p}$ with the dimensions:

$$\dim(\mathbb{C}^{n \times p}) = np \quad (12)$$

Obviously the algorithm complexity increases with the dimensions. As discussed before, optimization algorithms on manifolds work in an embedded or quotient space whose dimension will be much smaller than that of classical constrained optimization methods. Thus, optimization algorithms on manifolds not only have lower

complexity, but also perform better numerical properties. The corresponding algorithms on manifolds will be stated for IA in the next two subsections.

B. The Steepest Descent Algorithm on Complex Stiefel Manifold for IA

Informally, a manifold is a space that is “modeled on” Euclidean space. It can be defined as a subset of Euclidean space which is locally the graph of a smooth function.

Conceptually, the simplest approach to optimize a differentiable function is to continuously translate a test point in the direction of steepest descent on the constraint set until one reaches a point where the gradient is equal to zero. However, there are two challenges for optimization on manifolds. First, in order to define algorithms on manifolds, these operations above must be translated into the language of differential geometry. Second, once the test point shifts along the steepest descent direction, it must be retracted back to the manifold. Therefore, after reformulating the optimization problem on the Stiefel manifold, we introduce definitions about project operation and tangent space for retraction and gradient respectively.

In many cases, the underlying symmetry property can be exploited to reformulate the problem as a non-degenerate optimization problem on manifolds associated with the original matrix representation. Thus the constraint condition $\mathbf{V}^{[j]\dagger} \mathbf{V}^{[j]} = \mathbf{I}$ in the cost function (4), inspires us to solve the problem on the complex Stiefel manifold. The complex Stiefel manifold [17] $St(n, p)$ is the set satisfying

$$St(n, p) = \{\mathbf{X} \in \mathbb{C}^{n \times p} : \mathbf{X}^\dagger \mathbf{X} = \mathbf{I}\} \quad (13)$$

$St(n, p)$ naturally embeds in $\mathbb{C}^{n \times p}$ and inherits the usual topology of $\mathbb{C}^{n \times p}$. It is a compact manifold and from Proposition 3.3.3 in [18], we can get:

$$\dim(St(n, p)) = np - \frac{1}{2}p(p+1) \quad (14)$$

Another important definition is the projection. Assuming $\mathbf{Y} \in \mathbb{C}^{n \times p}$ is a rank- p matrix, the projection operator $\pi_{st}(\cdot) : \mathbb{C}^{n \times p} \rightarrow St(n, p)$ is given by

$$\pi_{st}(\mathbf{Y}) = \arg \min_{\mathbf{X} \in St(n, p)} \|\mathbf{Y} - \mathbf{X}\|^2 \quad (15)$$

It can be proved that there exists a unique solution if \mathbf{Y} has full column rank [18]. From (15), it can be acquired that the projection of an arbitrary rank- p matrix \mathbf{Y} onto the Stiefel manifold is defined to be the point on the Stiefel manifold closest to \mathbf{Y} in the Euclidean norm [19]. Besides, if the singular value decomposition (SVD) of \mathbf{Y} is $\mathbf{Y} = \mathbf{U} \Sigma \mathbf{V}^\dagger$, then

$$\pi_{st}(\mathbf{Y}) = \mathbf{U} \mathbf{I}_{n \times p} \mathbf{V}^\dagger \quad (16)$$

based on the Proposition 7 in [19].

Consider $\mathbf{X} \in St(n, p)$ and its disturbing point $\pi_{st}(\mathbf{X} + \varepsilon \mathbf{Y}) \in St(n, p)$ for certain directions matrix $\mathbf{Y} \in \mathbb{C}^{n \times p}$ and scalar $\varepsilon \in \mathbb{R}$. If \mathbf{Y} satisfies $f(\pi_{st}(\mathbf{X} + \varepsilon \mathbf{Y})) = f(\mathbf{X}) + O(\varepsilon^2)$ which means certain directions \mathbf{Y} do not cause $\pi_{st}(\mathbf{X} + \varepsilon \mathbf{Y})$ to move away from \mathbf{X} as ε increases. The collection of such

directions \mathbf{Y} is called the normal space at \mathbf{X} of $St(n, p)$ [18]. The tangent space $T_X(n, p)$ is defined to be the orthogonal complement of the normal space, which can be roughly illustrated as Fig. 2. And the mathematical expression of the tangent space $T_X(n, p)$ at $\mathbf{X} \in St(n, p)$ is defined by

$$T_X(n, p) = \{\mathbf{Z} \in \mathbb{C}^{n \times p} : \mathbf{Z} = \mathbf{X} \mathbf{A} + \mathbf{X}_\perp \mathbf{B}, \mathbf{A} \in \mathbb{C}^{p \times p}, \mathbf{A} + \mathbf{A}^\dagger = 0, \mathbf{B} \in \mathbb{C}^{(n-p) \times p}\} \quad (17)$$

in which $\mathbf{X}_\perp \in \mathbb{C}^{n \times (n-p)}$ is defined to be any matrix satisfying $[\mathbf{X} \ \mathbf{X}_\perp]^\dagger [\mathbf{X} \ \mathbf{X}_\perp] = \mathbf{I}$ and is the complement of $\mathbf{X} \in St(n, p)$. Also from [19], it can be obtained that the gradient of our cost function is in the tangent space $T_X(n, p)$. And the dimension of $T_X(n, p)$ is:

$$\dim(T_X(n, p)) = p(2n - p) \quad (18)$$

Obviously the steepest descent algorithm requires the com-

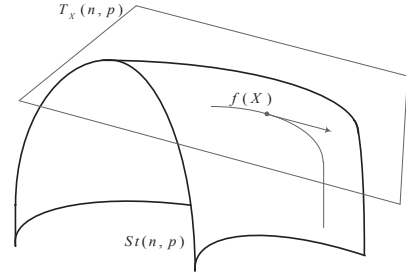


Fig. 2. Tangent space of Stiefel manifold

putation of the gradient. As we previously emphasize, the gradient is only defined after $T_X(n, p)$ is given an inner product:

$$\langle \mathbf{Z}_1, \mathbf{Z}_2 \rangle = \Re\{\text{tr}(\mathbf{Z}_2^\dagger (\mathbf{I} - \frac{1}{2} \mathbf{X} \mathbf{X}^\dagger) \mathbf{Z}_1)\} \quad (19)$$

where $\mathbf{Z}_1, \mathbf{Z}_2 \in T_X(n, p)$ and $\mathbf{X} \in St(n, p)$. The derivation of (19) can be found in [12]. Therefore, under the defined inner product, the steepest descent direction of the cost function $f(\mathbf{X})$ at the point $\mathbf{X} \in St(n, p)$ is

$$\mathbf{Z} = \mathbf{X} \mathbf{D}_X^\dagger \mathbf{X} - \mathbf{D}_X \quad (20)$$

where \mathbf{D}_X is the derivative of $f(\mathbf{X})$. The proof of (20) is in [19].

Inspired by [19] the proposed SD algorithm on complex Stiefel manifold is presented in **Algorithm 2**. From (19) and (20), it can be easily obtained that the inner product needed for the Armijo step rule is

$$\langle \mathbf{Z}^{[j]}, \mathbf{Z}^{[j]} \rangle = \Re\{\text{tr}(\mathbf{Z}^{[j]\dagger} (\mathbf{I} - \frac{1}{2} \mathbf{V}^{[j]} \mathbf{V}^{[j]\dagger}) \mathbf{Z}^{[j]})\} \quad (21)$$

which is used in Step 4 and 5; and the steepest descent on Stiefel manifold of our cost function is

$$\mathbf{Z}^{[j]} = \mathbf{V}^{[j]} \mathbf{D}_V^{[j]\dagger} \mathbf{V}^{[j]} - \mathbf{D}_V^{[j]} \quad (22)$$

which is used in Step 3. Noticing that the project operation $\pi_{st}(\cdot)$ in Step 6 (Step 4, 5) guarantees the newly computed solution $\mathbf{V}^{[j]}$ (or $\mathbf{B}_1^{[j]}, \mathbf{B}_2^{[j]}$) after iteration still satisfies $\mathbf{V}^{[j]} \in St(n, p)$. Using the method of SVD, we can easily compute the project operation.

Algorithm 2 The Steepest Descent Algorithm on Complex Stiefel Manifold for IA

Start with arbitrary precoder matrices $\mathbf{V}^{[1]}, \dots, \mathbf{V}^{[K]}$, set initial step size $\beta^{[j]} = 1$ and begin iteration.

for $j = 1 \dots K$

(1) Compute the Jacobian matrix $\mathbf{D}_R^{[j]}$ and $\mathbf{D}_I^{[j]}$

(2) Then get the derivative of f :

$$\mathbf{D}_V^{[j]} = (\mathbf{D}_R^{[j]} + i\mathbf{D}_I^{[j]})^T$$

(3) Get the steepest descent direction

$$\mathbf{Z}^{[j]} = \mathbf{V}^{[j]} \mathbf{D}_V^{[j]\dagger} \mathbf{V}^{[j]} - \mathbf{D}_V^{[j]}$$

(4) Compute $\mathbf{B}_1^{[j]} = \pi_{st}(\mathbf{V}^{[j]} + 2\beta^{[j]} \mathbf{Z}^{[j]})$,

if $f(\mathbf{V}^{[1]}, \dots, \mathbf{V}^{[K]}) - f(\mathbf{V}^{[1]}, \dots, \mathbf{V}^{[j-1]}, \mathbf{B}_1^{[j]}, \mathbf{V}^{[j+1]} \dots \mathbf{V}^{[K]}) \geq \beta^{[j]} \Re\{\text{tr}(\mathbf{Z}^{[j]\dagger}(\mathbf{I} - \frac{1}{2}\mathbf{V}^{[j]}\mathbf{V}^{[j]\dagger})\mathbf{Z}^{[j]})\}$, then set $\beta^{[j]} := 2\beta^{[j]}$, and repeat Step 4.

(5) Compute $\mathbf{B}_2^{[j]} = \pi_{st}(\mathbf{V}^{[j]} + \beta^{[j]} \mathbf{Z}^{[j]})$,

if $f(\mathbf{V}^{[1]}, \dots, \mathbf{V}^{[K]}) - f(\mathbf{V}^{[1]}, \dots, \mathbf{V}^{[j-1]}, \mathbf{B}_2^{[j]}, \mathbf{V}^{[j+1]} \dots \mathbf{V}^{[K]}) < \frac{1}{2}\beta^{[j]} \Re\{\text{tr}(\mathbf{Z}^{[j]\dagger}(\mathbf{I} - \frac{1}{2}\mathbf{V}^{[j]}\mathbf{V}^{[j]\dagger})\mathbf{Z}^{[j]})\}$, then set $\beta^{[j]} := \frac{1}{2}\beta^{[j]}$, and repeat Step 5.

(6) $\mathbf{V}^{[j]} = \pi_{st}(\mathbf{V}^{[j]} + \beta^{[j]} \mathbf{Z}^{[j]})$

(7) Continue till the cost function f is sufficiently small.

Discussion:

- (i) As previous stated, the algorithms in [11] and [12] are performed by moving the descent step along the geodesic of the constrained surface within each iteration. A disadvantage of this method is the redundant computational cost for calculating the path of a geodesic [19]. In this paper, we locally parameterize the manifold by Euclidean projection from the tangent space onto the manifold instead of moving along a geodesic, to achieve a modest reduction in the computational complexity of the algorithms.
- (ii) Recall (12) and (18), it can be obtained that when we reformulate the problem from $\mathbb{C}^{n \times p}$ to $St(n, p)$, the dimension of the optimization problem decreases from np to $np - \frac{1}{2}p(p+1)$. Although such dimension-dissension can be observed clearly, we still intend to reduce the dimensions of the space which the optimization algorithm works in. Thus the Grassmann manifold and its corresponding algorithm for IA are stated in the following subsection.

C. The Steepest Descent Algorithm on Complex Grassmann Manifold for IA

Notice that our cost function $f(\mathbf{V})$ satisfies $f(\mathbf{V}\mathbf{U}) = f(\mathbf{V})$ for any unitary matrix \mathbf{U} . Because

$$\begin{aligned} \mathbf{Q}^{[k]}(\mathbf{V}\mathbf{U}) &= \sum_{\substack{j=1 \\ j \neq k}}^K \frac{P^{[j]}}{d^{[j]}} \mathbf{H}^{[kj]} \mathbf{V}^{[j]} \mathbf{U}^{[j]} \mathbf{U}^{[j]\dagger} \mathbf{V}^{[j]\dagger} \mathbf{H}^{[kj]\dagger} \\ &= \sum_{\substack{j=1 \\ j \neq k}}^K \frac{P^{[j]}}{d^{[j]}} \mathbf{H}^{[kj]} \mathbf{V}^{[j]} \mathbf{I} \mathbf{V}^{[j]\dagger} \mathbf{H}^{[kj]\dagger} \\ &= \mathbf{Q}^{[k]}(\mathbf{V}) \end{aligned} \quad (23)$$

which means that multiplying unitary matrix \mathbf{U} does not change the eigenvalues and their corresponding eigenvectors of the interference covariance matrix at each receiver. Thus our cost function f should be minimized on the Grassmann manifold rather than on the Stiefel manifold. This is because the Grassmann manifold treats \mathbf{V} and $\mathbf{V}\mathbf{U}$ as equivalent points, leading to a further reduction in the dimension of the optimization problem. Similar with the previous section, we firstly introduce the definition about Grassmann manifold, then present the project operation and tangent space of Grassmann manifold for retraction and gradient respectively.

The complex Grassmann manifold $Gr(n, p)$ is defined to be the set of all p -dimensional complex subspaces of $\mathbb{C}^{n \times p}$. Grassmann manifold can be thought as a quotient space of the Stiefel manifold: $Gr(n, p) \simeq St(n, p)/St(p, p)$. Quotient space is more difficult to visualize, as it is not defined as set of matrices; rather, each point of the quotient space is an equivalence class of $n \times p$ matrices.

However, we can understand quotient space in this way: assuming $\mathbf{X} \in St(n, p)$ is a point on the Stiefel manifold, the columns of \mathbf{X} span an orthonormal basis for a p -dimensional quotient subspace. That is to say, if $[\mathbf{X}]$ denotes the subspace spanned by the columns of \mathbf{X} , then $\mathbf{X} \in St(n, p)$ implies $[\mathbf{X}] \in Gr(n, p)$. Therefore, there is a one-to-one mapping between points on the Grassmann manifold $Gr(n, p)$ and equivalence classes of $St(n, p)$. From (14), it can be acquired that:

$$\begin{aligned} \dim(Gr(n, p)) &= \dim(St(n, p)) - \dim(St(p, p)) \\ &= p(n - p) \end{aligned} \quad (24)$$

Let $\mathbf{Y} \in \mathbb{C}^{n \times p}$ be a rank- p matrix. The projection operator $\pi_{gr}(\cdot) : \mathbb{C}^{n \times p} \rightarrow Gr(n, p)$ onto the Grassmann manifold is defined to be

$$\pi_{gr}(\mathbf{Y}) = \left[\arg \min_{\mathbf{X} \in St(n, p)} \|\mathbf{Y} - \mathbf{X}\|^2 \right] \quad (25)$$

It also can be proved that there exists a unique solution if \mathbf{Y} has full column rank [18]. From (25), it can be acquired that the projection of an arbitrary rank- p matrix \mathbf{Y} onto the Grassmann manifold is defined to be the subspace spanned by the point on the Stiefel manifold closest to \mathbf{Y} in the Euclidean norm. Besides, if the QR decomposition of \mathbf{Y} is $\mathbf{Y} = \mathbf{Q}\mathbf{R}$,

the following equality holds:

$$\pi_{gr2}(\mathbf{Y}) = [\mathbf{Q}\mathbf{I}_{n \times p}] \quad (26)$$

The proof of (26) also can be found in [19]. From (26), it is obvious that $\pi_{gr}(\mathbf{Y})$ is the subspace spanned by the first p columns of \mathbf{Q} .

As discussed before, Grassmann manifold is a quotient space of the Stiefel manifold, thus its tangent space is a subspace of the Stiefel manifold's tangent space [18]. If $\mathbf{X} \in St(n, p)$, the tangent space $T_{[\mathbf{X}]}(n, p)$ at $[\mathbf{X}] \in Gr(n, p)$ of Grassmann manifold is:

$$T_{[\mathbf{X}]}(n, p) = \{\mathbf{Z} \in \mathbb{C}^{n \times p} : \mathbf{Z} = \mathbf{X}_\perp \mathbf{B}, \mathbf{B} \in \mathbb{C}^{(n-p) \times p}\} \quad (27)$$

Recall (18), the dimension of the tangent space $T_{[\mathbf{X}]}(n, p)$ of complex Grassmann manifold is:

$$\begin{aligned} \dim(T_{[\mathbf{X}]}(n, p)) &= \dim(T_X(n, p)) - \dim(T_X(p, p)) \\ &= p(2n - 2p) \end{aligned} \quad (28)$$

Besides the inner product of $T_{[\mathbf{X}]}(n, p)$ is given by:

$$\begin{aligned} \langle \mathbf{Z}_1, \mathbf{Z}_2 \rangle &= \Re\{tr(\mathbf{Z}_2^\dagger \mathbf{Z}_1)\}, \quad \mathbf{Z}_1, \mathbf{Z}_2 \in T_{[\mathbf{X}]}(n, p), \\ &\quad \mathbf{X} \in St(n, p) \end{aligned} \quad (29)$$

The derivation of (29) can be found in [12]. Therefore, under the defined inner product, the steepest descent direction [19] of the cost function $f(\mathbf{X})$ at the point $\mathbf{X} \in Gr(n, p)$ is:

$$\mathbf{Z} = -(\mathbf{I} - \mathbf{X}\mathbf{X}^\dagger)\mathbf{D}_X \quad (30)$$

where \mathbf{D}_X is the derivative of $f(\mathbf{X})$.

Algorithm 3 The Steepest Descent Algorithm on Complex Grassmann Manifold for IA

Start with arbitrary precoder matrices $\mathbf{V}^{[1]}, \dots, \mathbf{V}^{[K]}$, set initial step size $\beta^{[j]} = 1$ and begin iteration.

for $j = 1 \dots K$

(1) Compute the Jacobian matrix $\mathbf{D}_R^{[j]}$ and $\mathbf{D}_I^{[j]}$

(2) Then get the derivative of f :

$$\mathbf{D}_V^{[j]} = (\mathbf{D}_R^{[j]} + i\mathbf{D}_I^{[j]})^T$$

(3) Get the steepest descent direction

$$\mathbf{Z}^{[j]} = -(\mathbf{I} - \mathbf{V}^{[j]}\mathbf{V}^{[j]H})\mathbf{D}_V^{[j]}$$

(4) Compute $\mathbf{B}_1^{[j]} = \pi_{gr}(\mathbf{V}^{[j]} + 2\beta^{[j]}\mathbf{Z}^{[j]})$,

if $f(\mathbf{V}^{[1]}, \dots, \mathbf{V}^{[K]}) - f(\mathbf{V}^{[1]}, \dots, \mathbf{V}^{[j-1]}, \mathbf{B}_1^{[j]}, \mathbf{V}^{[j+1]}, \dots, \mathbf{V}^{[K]}) \geq \beta^{[j]}tr(\mathbf{Z}^{[j]H}\mathbf{Z}^{[j]})$, then set $\beta^{[j]} := 2\beta^{[j]}$, and repeat Step 4.

(5) Compute $\mathbf{B}_2^{[j]} = \pi_{gr}(\mathbf{V}^{[j]} + \beta^{[j]}\mathbf{Z}^{[j]})$,

if $f(\mathbf{V}^{[1]}, \dots, \mathbf{V}^{[K]}) - f(\mathbf{V}^{[1]}, \dots, \mathbf{V}^{[j-1]}, \mathbf{B}_2^{[j]}, \mathbf{V}^{[j+1]}, \dots, \mathbf{V}^{[K]}) < \frac{1}{2}\beta^{[j]}tr(\mathbf{Z}^{[j]H}\mathbf{Z}^{[j]})$, then set $\beta^{[j]} := \frac{1}{2}\beta^{[j]}$, and repeat Step 5.

(6) $\mathbf{V}^{[j]} = \pi_{gr}(\mathbf{V}^{[j]} + \beta^{[j]}\mathbf{Z}^{[j]})$

(7) Continue till the cost function f is sufficiently small.

The proposed SD algorithm on complex Grassmann man-

ifold is presented in **Algorithm 3**. Similar with the previous proposed algorithms, the Armijo step rule is performed to find a proper convergence step length. From (29) and (30), it can be easily concluded that the inner product needed for the Armijo step rule is

$$\langle \mathbf{Z}^{[j]}, \mathbf{Z}^{[j]} \rangle = tr(\mathbf{Z}^{[j]\dagger} \mathbf{Z}^{[j]}) \quad (31)$$

which is used in Step 4 and 5; and the steepest descent on Grassmann manifold of our cost function is

$$\mathbf{Z}^{[j]} = -(\mathbf{I} - \mathbf{V}^{[j]}\mathbf{V}^{[j]\dagger})\mathbf{D}_V^{[j]} \quad (32)$$

which is used in Step 3. And the project operation $\pi_{gr}(\cdot)$ in Step 6 (Step 4, 5) retracts the newly computed solution $\mathbf{V}^{[j]}$ (or $\mathbf{B}_1^{[j]}, \mathbf{B}_2^{[j]}$) back onto Grassmann manifold $Gr(n, p)$. Using QR decomposition, we can easily compute the project operation.

Discussion:

- Comparing $\dim(Gr(n, p)) = p(n - p)$ in (24) with $\dim(St(n, p)) = np - \frac{1}{2}p(p + 1)$ in (14), a further dimension reduction can be observed. Similarly, from (28) we can see another advantage of using the Grassmann manifold rather than the Stiefel manifold is that $T_{[\mathbf{X}]}(n, p)$ has only $p(2n - 2p)$ dimensions, whereas tangent space of $St(n, p)$ has $p(2n - p)$ dimensions. And from [20], it can be obtained that in our system model, if each transceiver is equipped with same amount of antenna ($M = N$), then

$$\sum_{k=1}^K d^{[k]} = K \cdot d = \frac{K \cdot M}{2} \quad (33)$$

and

$$d = \frac{M}{2} \quad (34)$$

Recall (12), (14) and (24), we can get that if M is large enough (M not only can represent the number of antennas each transceiver equipped, but also can refer to the number of time extension slots [1] [20]), hence

$$\frac{\dim(St(M, d))}{\dim(\mathbb{C}^{M \times d})} \approx \frac{3}{4} \quad (35)$$

which is a clear evidence for dimension-descension. And

$$\frac{\dim(Gr(M, d))}{\dim(\mathbb{C}^{M \times d})} = \frac{1}{2} \quad (36)$$

holds for any integer M . (36) means that optimization on Grassmann manifold would reduce dimension further. The trend of dimension-descension can be roughly illustrated in Fig. 3.

V. NUMERICAL RESULTS AND DISCUSSION

Without symbol extension, the feasible condition of k-user interference alignment [20] is given by:

$$M^{[j]} + N^{[j]} \geq (K + 1)d^{[j]} \quad (37)$$

For satisfying feasibility and simple computation, we consider a 3-user 2×2 MIMO interference channel where the desired

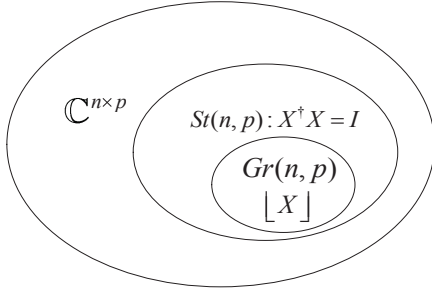


Fig. 3. Trend of dimension-descension

DoF per user $d^{[j]}$ is 1. All the algorithms are executed under the same scenario including randomly generated channel coefficients, initial precoder matrices and convergence step length. We simulated the proposed three SD algorithms through 100 simulation realizations.

As shown in Fig. 4, Fig. 5 and Fig. 6, each curve represents an individual simulation realization and all results converge after 20 or more iterations, which is a clear indication of the algorithm convergence performance in such limited iterations. In order to compare the convergence performance, the average values of 100 realizations results are illustrated in Fig. 7. It can be observed that the algorithms on manifolds have better convergence performance comparing with the classical optimization method as our expectation.

This is attributed to the reason that we reformulate the constrained optimization problem to an unconstrained one on manifolds with lower complexity and better numerical properties; then locally parameterize the manifolds by a Euclidean projection of the tangent space on to the manifolds instead of moving along the geodesic, as stated in the previous sections. Moreover the convergence performance curve of SD algorithm on Stiefel manifold and the curve of SD algorithm on Grassmann manifold almost overlap. Recall that optimization on Grassmann manifold would reduce dimension further, therefore the SD algorithm on Grassmann manifold will guarantee performance and reduce the computation complexity at the same time.

Meanwhile, since there are two interference signals at each receiver, as shown in Fig. 8, the angles between the spaces spanned by each interference signals asymptotically converge to zero within one simulation realization, which is another evidence for achieving the perfect interference alignment.

Finally we compare the system sum-rate of the proposed algorithms. Fig. 9 shows that the SD algorithm on Stiefel manifold and the SD algorithm on Grassmann manifold almost have the same performance, and outperform the other classical optimization algorithms. More importantly, at high SNR the DoF of the three proposed algorithms nearly achieve 3, which is the maximum theoretical value ($KM/2 = 3$). Therefore the perfect interference alignment is successfully achieved.

Two reasons leading to the fact that the algorithms on manifolds obtain higher system capacity are presented below:

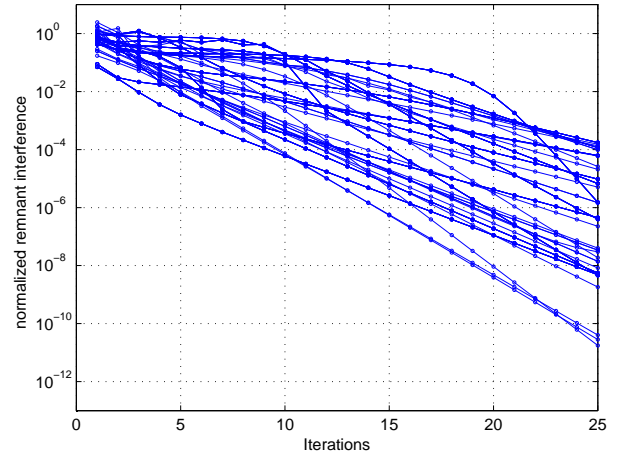


Fig. 4. Normalized remnant interference of the SD algorithm in complex space

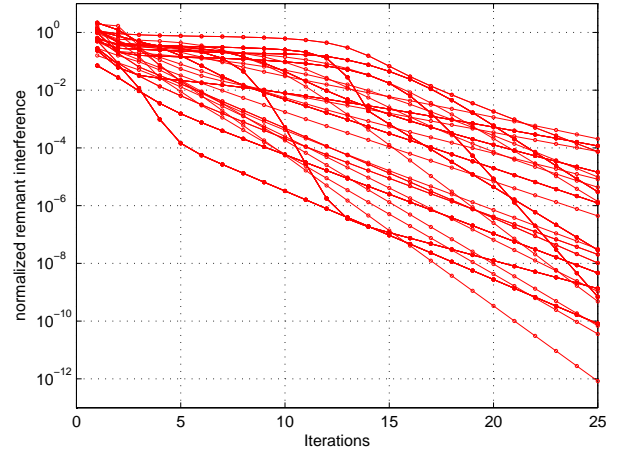


Fig. 5. Normalized remnant interference of the SD algorithm on Stiefel manifold

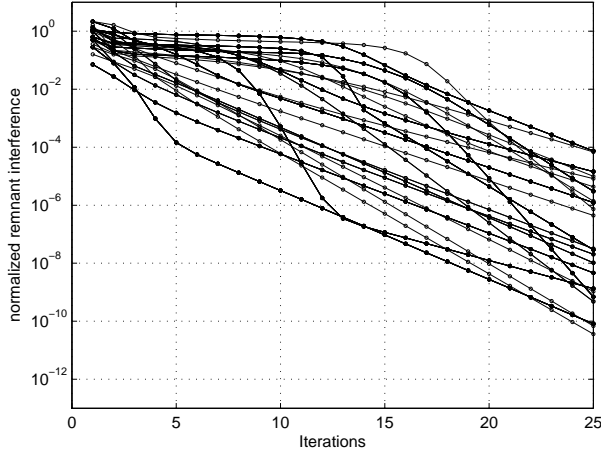


Fig. 6. Normalized remnant interference of the SD algorithm on Grassmann manifold

- (i) It is noticed that our cost function actually is the interference power spilled from the interference space to the desired signal space. With the better convergence performance, the SD algorithms on manifolds will have less remnant interference in the desired signal space within same iteration times. Therefore, the SD algorithms on manifolds will get higher SINR [5]:

$$SINR = \frac{\text{signal power}}{\text{noise} + \text{remnant interference}} \quad (38)$$

which leads to high capacity.

- (ii) At each receiver, the zero forcing filter is adopted. It will project the desired signal power and the remnant interference onto the subspace which is orthogonal with the subspace spanned by the interference. After performing the SD algorithms on manifolds, it is observed that in the Euclidean norm distance, the subspace spanned by desired signal is more close to the orthogonal complement of the interference subspace. Therefore, even the proposed algorithms on manifolds finally get the same remnant interference as the classical optimization methods results. The algorithms on manifolds will suffer from less power lose during the projection operated by zero forcing filter, hence higher system capacity can be achieved.

We notice that better throughputs may be attained by using non-unitary precoders, or by applying power water-filling in the equivalent non-interfering MIMO channels. Nevertheless, these methods to increase throughputs can be performed as the second step after the interference alignment is achieved [11]. Thus in this paper, we only need to concentrate on the first step to find the perfect solutions of interference alignment.

VI. CONCLUSION AND FUTURE WORK

In this paper, we focus on the precoder designs on manifolds for interference alignment. By restricting the optimization only

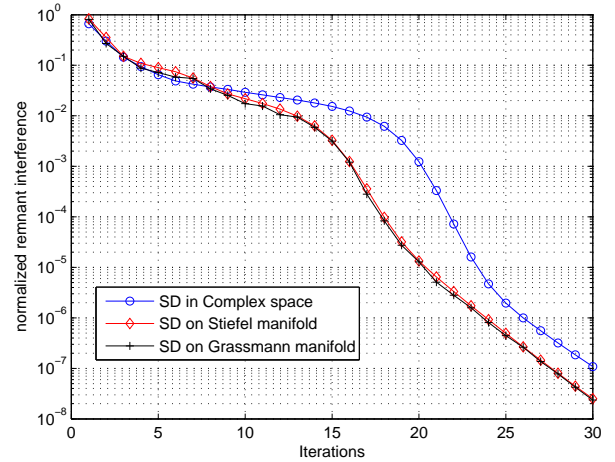
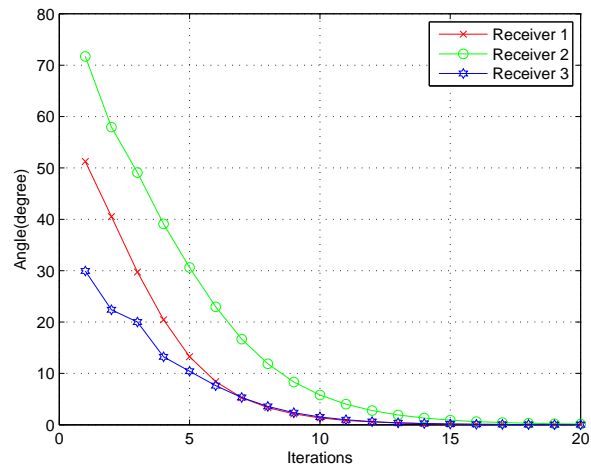


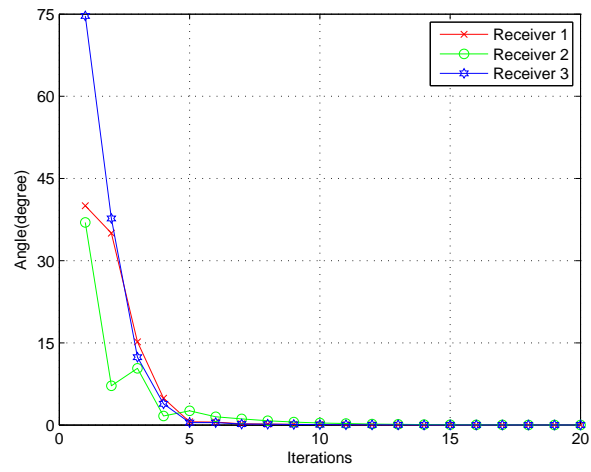
Fig. 7. Convergence performance

at the transmitters' side, it will alleviate the significant overhead induced by alternation between the forward and reverse links. A classical SD algorithm in multi-dimensional complex space is proposed first. Then we reform the optimization problem on Stiefel manifold, and propose a novel SD algorithm on this manifold with lower dimensions. Moreover, aiming at further reducing the complexity, Grassmann manifold is introduced to derive corresponding algorithm for reaching the perfect interference alignment. Numerical simulations show that comparing with previous methods, the proposed algorithms on manifolds have better convergence performance and higher system capacity, also achieve the maximum DoF.

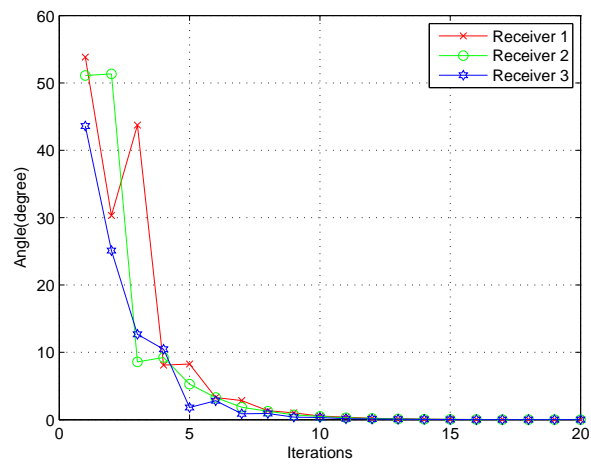
The proof for convergence of the proposed algorithms is quite simple. Our cost function is non-negative with the low bound zero. It monotonically decreases within each iteration. Therefore it must converge to a solution which is very close to zero. However, there is no guarantee that our cost function



(a) SD algorithm in complex space



(b) SD algorithm on Stiefel manifold



(c) SD algorithm on Grassmann manifold

Fig. 8. Angles between interfering spaces at each receiver

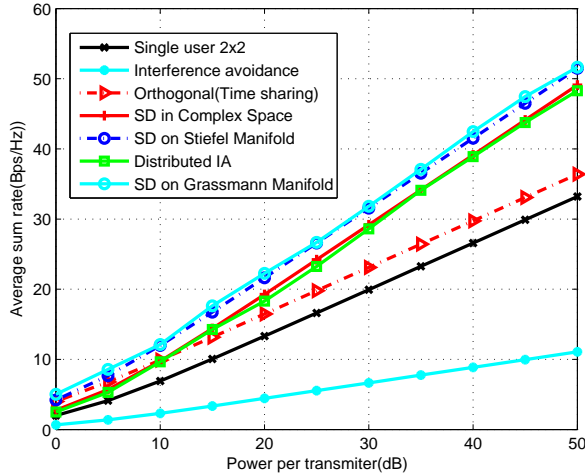


Fig. 9. Sum-rate capacity

is convex [5]. Thus finding a global optimum is in our future work.

We also do some simulations by using more sophisticated algorithm, such as Newton-type method, to achieve quadratic convergence. Yet, except for the increased computational complexity, the Newton method will converge to the closet critical point [19]. Therefore the Newton method coupled with the steepest descent algorithm (a few iterations of the SD algorithm are performed first to move close to a local minimum before the Newton algorithm is applied) will be investigated in our future work too.

ACKNOWLEDGMENT

This work was supported by National Natural Science Foundation of China (No. 61171112) and MIIT of China (No. 2010ZX03005-001-02).

REFERENCES

- [1] V. R. Cadambe and S. A. Jafar, "Interference alignment and degrees of freedom of the K-user interference channel," *IEEE Trans. Information Theory*, vol. 54, no. 8, pp. 3425-3441, Aug. 2008.
- [2] M. Maddah-Ali, A. Motahari, and A. Khandani, "Signaling over MIMO multi-base systems: Combination of multi-access and broadcast schemes," *Int. Symp. Inform Theory 2006*, pp. 2104-2108.
- [3] M. A. Maddah-Ali, A. S. Motahari, A. K. Khandani, "Communication over MIMO X channels: interference alignment, decomposition, and performance analysis," *IEEE Transactions on Information Theory*, vol. 54, no. 8, pp. 3457-3470, Aug. 2008.
- [4] Cadambe, Jafar, "Reflections on interference alignment and the degrees of freedom of the K-user MIMO interference channel," *IEEE Inform Theory Society Newsletter*, vol. 54, no. 4, pp. 5-8, Dec 2009.
- [5] K. S. Gomadam, V. R. Cadambe, and S. A. Jafar, "Approaching the capacity of wireless networks through distributed interference alignment," *IEEE GLOBECOM*, New Orleans, Louisiana, Nov 2008, pp. 1-6.
- [6] S. W. Peters, R. W. Heath, "Interference alignment via alternating minimization," *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, Taipei, Taiwan, April 2009, pp. 2445-2448.
- [7] J. Raj Kumar and F. Xue, "An iterative algorithm for joint signal and interference alignment," *IEEE ISIT 2010*, Austin, pp. 2293 - 2297.
- [8] Sheng Liu and Yinggang Du and Meng Zhao, "A new joint iterative transceiver design with interference alignment method," *Wireless and Optical Communications Conference, 19th Annual*, Shanghai, 2010, pp. 1-5.
- [9] Hui Shen and Bin Li, "A novel iterative interference alignment scheme via convex optimization for the MIMO interference channel," *Vehicular Technology Conference*, Ottawa, Set. 2010, pp. 1-5.
- [10] Hui Shen and Bin Li and Meixia Tao and Yi Luo, "The new interference alignment scheme for the MIMO interference channel," *Wireless Communications and Networking Conference*, Sydney, Apr. 2010, pp. 1-6.
- [11] I. Santamaria, O. Gonzalez, R. W. Heath, and S. W. Peters, "Maximum sum-rate interference alignment algorithms for MIMO channels," *IEEE GLOBECOM 2010*, pp. 1-6, Dec. 2010.
- [12] A. Edelman, T. A. Arias, and S. T. Smith, "The geometry of algorithms with orthonormality constraints," *SIAM J. Matrix Anal. Applic.*, vol. 20, no. 2, pp. 303 - 353, 1998.
- [13] Hadi G. Ghauch, Constantinos B. Papadias, "Interference alignment: A one-sided approach," *IEEE Global Telecommunications Conference*, Texas, Dec. 2011, pp. 1-5.
- [14] A. Hjørungnes, *Complex-Valued Matrix Derivatives: With Applications in Signal Processing and Communications*. Cambridge, UK: Cambridge University Press, 2011.
- [15] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, UK: Cambridge University Press, 2004.
- [16] E. Polak, *Optimization: Algorithms and Consistent Approximations*, New York: Springer-Verlag, 1997.
- [17] Xianda Zhang, *Matrix Analysis and Applications*. Beijing, China: Springer, 2004.
- [18] P. A. Absil, R. Mahony and R. Sepulchre, *Optimization Algorithms on Matrix Manifolds*. New Jersey: Princeton University Press, 2008.
- [19] J. H. Manton, "Optimization algorithms exploiting unitary constraints," *IEEE Trans. Signal Process.* 50 (2002), no. 3, 635-650.
- [20] Syed A. Jafar, "Interference Alignment: A New Look at Signal Dimensions in a Communication Network," *Foundations and Trends in Communications and Information Theory*, Vol. 7, No. 1, pages: 1-136, 2011.

